

```
#!/bin/sh

# $Header: available_media.sh,v 1.5.94.1 2008/02/08 00:11:03 $

#
#*****
#* $VRTScprght: Copyright 1993 - 2008 Symantec Corporation, All Rights Reserved
$
#*****
#
# available_media
#
# This script can be used by an administrator as a convenient way to
# list ALL NetBackup volumes in the Media Manager volume database, and
# augment that list with information on the volumes currently "assigned"
# to NetBackup.
# The major benefit of this script is that it eliminates the need to track
# information from multiple databases and do the correlation manually.
#
# This script can be used in a crontab entry to get periodic reports of
# available media.
#
# caveat:
#     this script assumes that the Media Manager database and NetBackup
#     media database are both resident on the host where this script is run.
#
#
=====
=
get_med_type()
{
```

```
if [ "${vmediatype}" = "REWR_OPT" ]
then
    vmediatype="REWR_OPT"
elif [ "${vmediatype}" = "WORM_OPT" ]
then
    vmediatype="WORM_OPT"
elif [ "${vmediatype}" = "T120" ]
then
    vmediatype="T120    "
elif [ "${vmediatype}" = "8MM" ]
then
    vmediatype="8MM     "
elif [ "${vmediatype}" = "8MM_CLN" ]
then
    vmediatype="8MM_CLN "
elif [ "${vmediatype}" = "HCART" ]
then
    vmediatype="HCART   "
elif [ "${vmediatype}" = "HREEL" ]
then
    vmediatype="HREEL   "
elif [ "${vmediatype}" = "QCART" ]
then
    vmediatype="QCART   "
elif [ "${vmediatype}" = "4MM" ]
then
    vmediatype="4MM     "
elif [ "${vmediatype}" = "4MM_CLN" ]
then
    vmediatype="4MM_CLN "
```

```
elif [ "${vmediatype}" = "DLT" ]
then
    vmediatype="DLT    "
elif [ "${vmediatype}" = "DLT_CLN" ]
then
    vmediatype="DLT_CLN "
elif [ "${vmediatype}" = "HC_CLN" ]
then
    vmediatype="HC_CLN  "
elif [ "${vmediatype}" = "HCART2" ]
then
    vmediatype="HCART2  "
elif [ "${vmediatype}" = "HC2_CLN" ]
then
    vmediatype="HC2_CLN "
elif [ "${vmediatype}" = "HCART3" ]
then
    vmediatype="HCART3  "
elif [ "${vmediatype}" = "HC3_CLN" ]
then
    vmediatype="HC3_CLN "
elif [ "${vmediatype}" = "DLT2" ]
then
    vmediatype="DLT2    "
elif [ "${vmediatype}" = "DLT2_CLN" ]
then
    vmediatype="DLT_CLN2"
elif [ "${vmediatype}" = "DLT3" ]
then
    vmediatype="DLT3    "
```

```
elif [ "${vmediatype}" = "DLT3_CLN" ]
then
    vmediatype="DLT_CLN3"
elif [ "${vmediatype}" = "8MM2" ]
then
    vmediatype="8MM2    "
elif [ "${vmediatype}" = "8MM2_CLN" ]
then
    vmediatype="8MM2_CLN"
elif [ "${vmediatype}" = "D2" ]
then
    vmediatype="D2      "
elif [ "${vmediatype}" = "D2_CLN" ]
then
    vmediatype="D2_CLN  "
elif [ "${vmediatype}" = "DTF" ]
then
    vmediatype="DTF     "
elif [ "${vmediatype}" = "DTF_CLN" ]
then
    vmediatype="DTF_CLN "
elif [ "${vmediatype}" = "8MM3" ]
then
    vmediatype="8MM3    "
elif [ "${vmediatype}" = "8MM3_CLN" ]
then
    vmediatype="8MM3_CLN"
fi
}
#
```

=====

=

```
if [ "$1" = "-usage" ]
```

```
then
```

```
    echo "usage: $0 [-summary]"
```

```
    exit 0
```

```
fi
```

```
VMQUERY=/usr/opensv/volmgr/bin/vmquery
```

```
VMPOOL=/usr/opensv/volmgr/bin/vmpool
```

```
BPMEDIALIST=/usr/opensv/netbackup/bin/admincmd/bpmedialist
```

```
if [ -f /usr/opensv/netbackup/version ]
```

```
then
```

```
    HARDWARE=`head -1 /usr/opensv/netbackup/version | cut -f2 -d" "`
```

```
else
```

```
    echo "/usr/opensv/netbackup/version not found"
```

```
    exit 1
```

```
fi
```

```
TR=/usr/bin/tr
```

```
case "$HARDWARE" in
```

```
    SUN4)
```

```
        TR=/bin/tr
```

```
        ;;
```

```
    RS6000)
```

```
        TR=/bin/tr
```

```
        ;;
```

```
SOLARIS)

    TR=/bin/tr

    ;;

esac

/usr/opensv/netbackup/bin/admincmd/bpauthorize -test_admin >/dev/null
status=$?

if [ ${status} -ne 0 ] ; then

    /bin/echo ""

    /bin/echo "You do not have the proper authorization to perform"

    /bin/echo "this task. Please resolve and try again."

    exit ${status}

fi

/bin/rm -rf /tmp/avail_media_output
/bin/rm -rf /tmp/avail_media_sorted_output
/bin/touch /tmp/avail_media_output

TAB=`/bin/echo ' ' | $TR ' ' '\011'`

VMPOOL_OUTPUT=/tmp/vmool_output
/bin/rm -rf $VMPOOL_OUTPUT
touch $VMPOOL_OUTPUT

$VMPOOL -listall -b 2>/dev/null | tail +3 | sort +0 1>$VMPOOL_OUTPUT 2>/dev/null

BPMEDIALIST_OUTPUT=/tmp/bpmedialist_output
/bin/rm -rf $BPMEDIALIST_OUTPUT
touch $BPMEDIALIST_OUTPUT
```

```
$BPMEDIA LIST -m list -l 1>$BPMEDIA LIST_OUTPUT 2>/dev/null
```

```
VMQUERY_OUTPUT=/tmp/vmquery.out
```

```
ASTERISK_DISPLAYED=/tmp/asterisk_displayed
```

```
cat $VMPOOL_OUTPUT |
```

```
while read poolname poolhost pooluser poolgroup pooldesc
```

```
do
```

```
    /bin/rm -rf $VMQUERY_OUTPUT
```

```
    /bin/rm -rf /tmp/avail_media_output
```

```
    touch $VMQUERY_OUTPUT
```

```
    $VMQUERY -pn $poolname -bx 2>/dev/null | tail +4 1>>$VMQUERY_OUTPUT
```

```
2>/dev/null
```

```
cat $VMQUERY_OUTPUT |
```

```
while read vmediaid vmediatype vrobottype vrobotnum vrobotslot vside vvol vop  
vmnts vlmtdate vlmttime vasgndate vasgntime vpool
```

```
do
```

```
    /bin/rm -rf /tmp/bpmediaid_found
```

```
    if [ "${vasgndate}" = "---" -a "${vasgntime}" = "---" ]
```

```
    then
```

```
        bpkbytes="Unknown"
```

```
        bpretlev="-"
```

```
        bpstatus=AVAILABLE
```

```
        get_med_type
```

```
        /bin/echo "$poolname${TAB}$vmediaid${TAB}$vmediatype
```

```
$vrobottype${TAB} $vrobotnum${TAB} $vrobotslot${TAB} $vside${TAB} $bpretlev
```

```
$bpkbytes${TAB}$bpstatus" >>/tmp/avail_media_output
```

```
else
```

```
grep "^$vmediaid" $BPMEDIALIST_OUTPUT |
```

```
while read bpmediaid bppartner bpver bpden bpalloc bplwrite bpexp  
bplrest bpkbytes bpnimages bpvimages bpretlev bpunused1 bpnumrest bpstat bprest
```

```
do
```

```
  /bin/touch /tmp/bpmediaid_found
```

```
  bpstatus=""
```

```
  spaces="      "
```

```
  isencrypted=0
```

```
  isbemedia=0
```

```
  isworm=0
```

```
  ismpx=0
```

```
  # Translate bpstat into a string
```

```
  if [ "$bpstat" -gt 262143 ]
```

```
  then
```

```
    # Ignore this bit.
```

```
    bpstat=`expr $bpstat - 262144`
```

```
  fi
```

```
  if [ "$bpstat" -gt 131071 ]
```

```
  then
```

```
    # Ignore this bit.
```

```
    bpstat=`expr $bpstat - 131072`
```

```
  fi
```



```
if [ "$bpstat" -gt 65535 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 65536`
fi

if [ "$bpstat" -gt 32767 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 32768`
fi

if [ "$bpstat" -gt 16383 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 16384`
fi

if [ "$bpstat" -gt 8191 ]
then
    isencrypted=1
    bpstat=`expr $bpstat - 8192`
fi

if [ "$bpstat" -gt 4095 ]
then
    isbemedias=1
    bpstat=`expr $bpstat - 4096`
fi
```

```
if [ "$bpstat" -gt 2047 ]
then
    isworm=1
    bpstat=`expr $bpstat - 2048`
fi

if [ "$bpstat" -gt 1023 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 1024`
fi

if [ "$bpstat" -gt 511 ]
then
    ismpx=1
    bpstat=`expr $bpstat - 512`
fi

if [ "$bpstat" -gt 255 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 256`
fi

if [ "$bpstat" -gt 127 ]
then
    bpstat=`expr $bpstat - 128`
    bpstatus=IMPORTED
fi
```

```
if [ "$bpstat" -gt 63 ]
then
    bpstat=`expr $bpstat - 64`
    # Add * on retention level to indicate
    # this media id has multiple retention
    # levels on it. Reduce the number of
    # spaces to account for the *.
    bpretlev=${bpretlev}*
    spaces="    "
    touch $ASTERISK_DISPLAYED
fi

if [ "$bpstat" -gt 31 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 32`
fi

if [ "$bpstat" -gt 15 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 16`
fi

if [ "$bpstat" -gt 7 ]
then
    bpstat=`expr $bpstat - 8`
    if [ X"${bpstatus}" = X ]
    then
```

```
        bpstatus=FULL
    else
        bpstatus=${bpstatus}/FULL
    fi
fi

if [ "$bpstat" -gt 3 ]
then
    # Ignore this bit.
    bpstat=`expr $bpstat - 4`
fi

if [ "$bpstat" -gt 1 ]
then
    bpstat=`expr $bpstat - 2`
    if [ X"${bpstatus}" = X ]
    then
        bpstatus=SUSPENDED
    else
        bpstatus=${bpstatus}/SUSPENDED
    fi
fi

if [ "$bpstat" -gt 0 ]
then
    if [ X"${bpstatus}" = X ]
    then
        bpstatus=FROZEN
    else
        bpstatus=${bpstatus}/FROZEN
    fi
fi
```

```
        fi
    fi

    if [ X"${bpstatus}" = X ]
    then
        if [ $ismpx -eq 1 ]
        then
            bpstatus=ACTIVE/MPX
        else
            bpstatus=ACTIVE
        fi
    else
        if [ $ismpx -eq 1 ]
        then
            bpstatus=${bpstatus}/MPX
        fi
    fi

    if [ $isworm -eq 1 ]
    then
        bpstatus=${bpstatus}/WORM
    fi

    if [ $isbimedia -eq 1 ]
    then
        bpstatus=${bpstatus}/BEMEDIA
    fi

    if [ $isencrypted -eq 1 ]
    then
```

```
        bpstatus=${bpstatus}/ENCRYPTION
    fi

    get_med_type

        /bin/echo "$poolname${TAB}$vmediaid${TAB}$vmediatype
$vrobottype${TAB} $vrobotnum${TAB} $vrobotslot${TAB} $vside${TAB}
$bpretlev$spaces$bpkbytes${TAB}$bpstatus" >>/tmp/avail_media_output

done

if [ ! -f /tmp/bpmediaid_found ]
then
    bpkbytes="Unknown"
    bpretlev="-"
    bpstatus=DBBACKUP

    get_med_type

        /bin/echo "$poolname${TAB}$vmediaid${TAB}$vmediatype
$vrobottype${TAB} $vrobotnum${TAB} $vrobotslot${TAB} $vside${TAB} $bpretlev
$bpkbytes${TAB}$bpstatus" >>/tmp/avail_media_output

    fi
fi

done

#/bin/echo "$poolname pool" >>/tmp/avail_media_sorted_output
# /bin/echo "" >>/tmp/avail_media_sorted_output
```

```

# Sort the entries on media id.

if [ -f /tmp/avail_media_output ]
then
    cat /tmp/avail_media_output | sort +8 +0 >>
/tmp/avail_media_sorted_output
#     /bin/echo "" >>/tmp/avail_media_sorted_output
fi
done

# /bin/echo
"Pool${TAB}media${TAB}media${TAB}robot${TAB}robot${TAB}robot${TAB}side/${TAB}
}ret    size${TAB}status"

# /bin/echo "Name${TAB} ID${TAB} type${TAB}type${TAB}
# ${TAB}slot${TAB}face${TAB}level    KBytes"

# /bin/echo
"-----
-----"

/bin/cat /tmp/avail_media_sorted_output | awk '{print $1,""$2,""$9,""$10}'

/bin/rm -rf /tmp/avail_media_sorted_output /tmp/avail_media_output

$VMQUERY_OUTPUT $VMPOOL_OUTPUT $BPMEDIALIST_OUTPUT

```